Anirudh Dhebar
Sloan School of Management
Massachusetts Institute of Technology
38 Memorial Drive, E56-329
Cambridge, Massachusetts 02142-1347

Telephone: (617) 253-5056
e-mail address: adhebar@mit.edu

Information Technology and Product Policy (A): "Smart" Products

Anirudh Dhebar

August 1995 **WP # 3855**

ICRMOT WP # 133-95

Information Technology and Product Policy (A): "Smart" Products

Abstract

Increasingly, information technology (IT) is incorporated in products to make them "smart"—to provide the user improved information about and control over performance, greater automation, and enhanced features, functions, and capabilities. These product improvements—and one hopes the consumer sees them as improvements—are made possible by the programming capabilities of microprocessors and other electronic devices, which expand the set of benefit-enhancing attributes that can be designed into a product and make it easier and cheaper to change, add, or drop a growing set of attributes. This article outlines three major product-policy concerns stemming from the new-found design facility, flexibility, and economy: getting the product and product line "right," managing the speed and nature of product change, and establishing product-use standards. A central message that emerges is the need for a new system of checks and balances to restrain the product supplier from piling on the features, too fast and with dysfunctional disruptions in product-use standards. The message should be of interest to a broad set of organizational functions: product design and development, product and marketing management, and IT management.

Mention "information technology," and many managers will think of computers—as nerve centers for an organization's information, control, decision-support, and communications systems; as enablers of task automation; as foci of new product-design and manufacturing technologies; as coordinators of material flows and logistical support; and so on. While these information-technology (IT) applications can—and do—affect a firm's products, the impact usually is indirect.[1]

This series of articles focuses on IT applications that have a direct impact on products and product lines. I consider two settings. In this article, I look at "smart" products—physical products that have IT incorporated in them. In an accompanying article (Dhebar, 1995d), I take up the case of information products, the "production" (information collection, storage, editing, and publishing) and distribution of which are increasingly dependent on IT.

IT in Products: How and to What Effect

> Craig Fields, chairman of Microelectronics and Computer Technology Corporation, once predicted that by the turn of the century, General Motors would be selling more computer power than IBM. It remains a reasonable forecast. Whereas a typical car in 1970 contained about $75 worth of electronics, most of it in a radio, the average today is about $2,000. And on such cars as Ford's New Lincoln Continental, laden with gadgets such as automatic seat-tilters, radio-tuners and suspension-adjusters, all triggered by a transmitter in the driver's key ring, the figure tops $3,000. (*The Economist*, 1995: 76)

It is not only cars that are getting smart. The last two decades have witnessed an explosion in the use of electronics, in particular, microprocessors, in toys, entertainment systems, household

[1]The impact, though indirect, may still be significant. Better and more timely information, improved decision support, design automation, flexible manufacturing, electronic data interchange, and so on, can have a significant impact on the product line. I focus on direct IT effects in this article not on the basis of their significance or otherwise but because of the specific product policy issues that are raised.

appliances, office and factory equipment, instrumentation devices, telecommunications equipment, transportation systems, and so on. The move to greater intelligence is not limited to individual products; there is talk, too, of smart homes, offices, highways, and communities. The social implications of all this smartness, while difficult to predict, are likely to be significant and pervasive. This article, however, is not about the social implications of increasing product smartness; my concern is more prosaic: I want to explore the consequences for product policy.

Generally speaking, the use of microprocessors and other electronic devices in a product can result in three types of benefits for the user:

- improved information about and control over product performance;
- greater automation of product functions; and
- enhanced product features, functions, and capabilities.

Take the radio receiver. Time was when "playing the radio" required four actions, all manual: switching the radio on, selecting a band, setting the desired station, and adjusting the volume. The same three tasks continue to be at the heart of radio use, but today's radio receivers can do much more—and the same differently. A glance at the controls reveals that the analog readout is gone, as is the pointer traversing the spectrum of frequencies. Instead, a digital alphanumeric display tells the time, the mode (TUNER, TAPE, CD, and so on), the station frequency, the acoustic properties of the sound, and so on. To tune to a station, the listener does not rotate the tuning dial; instead he or she pushes a series of buttons. Better still, frequently listened-to stations can be programmed into memory and recalled at the push of a button. Before, to change stations or to adjust the volume, the listener had to get up and physically interact with the radio receiver; now, most controls are accessible through a portable remote-control device. Finally, today's receiver not only receives radio signals, it also doubles as the command center for a complete entertainment system, lulling the listener to sleep with classical music on one frequency, taping a talk show on another frequency, and waking the listener with news on yet another frequency—all with digital precision.

The three "benefits"[2]—improved information about and control over product performance, greater automation, and enhanced functionality—are made possible by the programming capabilities of microprocessors and other electronic devices. The liberating influence of these capabilities on product design cannot be overstated and has at least three important facets:

- the set of attributes that can be designed into a product often is much larger,
- certain attributes are more easily designed in and changed over time, and
- the cost of changing, adding, or dropping attributes is often significantly lower.

The last point is illustrated by the following quote:

> ... Modern electronics has turned the economics of design on its head. No more does the cost of adding features limit the number of capabilities a designer can put into a machine. The chip that was designed to perform a single basic function can frequently be made to do 2, 3, 4, or 50 operations at negligible cost—so why not pile on the features? (*Business Week*, 1991: 58)

The *Economist* article on smart cars makes a similar point:

> ... Adapting ABS [anti-lock braking systems] and computerized engine control into a smart traction system involves little physical change to a car—it is almost entirely a matter of software, giving the car's computer a new set of instructions. It adds only another $25 or so to the $200 it costs a car company to install ABS, but even so, companies which offer traction control charge several hundred dollars extra for it. (*The Economist*, 1995: 76-78)[3]

---

[2]The reader still struggling with "How do I set the clock in my VCR?" may question the use of the word "benefits." The point is well taken and I touch on it later in the article.

[3]The shift from physical to software-based change as the enabling mechanism for product improvement raises interesting questions about the staffing of the product design and development function: quite possibly, very different skills may be needed for realizing software-based product changes than hardware-based ones, and people proficient in the latter may not be good at the former.

The design facility, flexibility, and economy resulting from the use of microprocessors and other electronic devices have potentially major implications for the product line a firm is able to offer at any point in time, the pace and nature of product change the firm is able to implement over time, and the amount of product experimentation the firm is able to carry out in the market.

Take the wristwatch. In the past, product variety in wristwatches was implemented more through differences in appearance than through differences in functions. True, there were manual-winding and self-winding watches and watches with and without date-and-day displays and alarm-and-chronograph functions, but, otherwise, different models were distinguished by their size, strap, face, and so on. Contrast that with the relatively inexpensive, black, plastic, digital wristwatches of the late 1970s-early 1980s. Most of these watches looked similar, but there was great variety in features and functions—variety achieved through variations in the programming of the microprocessor. Not only were there many different models on offer, it was relatively easy and cheap to add and drop features and there was rapid churning of the product line.

The greater facility, flexibility, and economy that IT brings to product design do not come without a price, and therein lie some critical product-policy challenges. The central concerns that emerge—getting the product and the product line "right," managing the speed and nature of product change, and establishing product-use standards—are not unique to IT-based products but, nevertheless, merit special attention and are the subject of the following discussion.

I have two goals for the article: (1) to present a coherent articulation of the more important product policy questions that arise because of the use of IT in products; and (2) to draw the attention of product managers, marketing managers, product development managers, IT managers, product designers, and senior management to the same. With the wider audience in mind, I have chosen to survey the critical issues rather than to provide an in-depth treatment of any one or more of them.

The choice is only partly deliberate. The traditional literature on product policy does not distinguish between products with or without IT and the existing IT literature tends to focus on situations where IT's impact is indirect (see opening paragraph)—and, references cited in this article

notwithstanding, in-depth answers are not available to some of the questions that are raised. Hopefully, this article will sensitize managers and researchers to the important questions and impart momentum to the effort to get to the answers.

Getting the Product and the Product Line Right

As suppliers incorporate IT in products, perhaps the greatest challenge lies in being judicious about the use of the technology: IT must be employed to offer the most appropriate information about and control over product performance, to deliver the optimal level of automation, and to enable the best set of features, functions, and capabilities. Inadequate information about and control over product performance, too little automation, and restricted functionality, and the supplier may lose the opportunity to add value to the consumer, capture market share, stimulate new demand, and encourage existing customers to repurchase the product or to purchase a "new-and-improved" version. On the other hand, too much information about and control over product performance, excessive automation, and an overwhelming set of features, and too many consumers may be befuddled by the product. Precisely because IT brings added facility, flexibility, and economy to design, the risk is significant of falling into the "too much," "excessive," and "overwhelming" camp:

> ... If economics doesn't restrain designers, designers must restrain themselves—a wholly new design discipline born of the electronic age. "Who in the world programs three weeks of TV ahead of time?" ... (*Business Week*, 1991: 62)

How should designers restrain themselves? The obvious—but, given the giddying temptation of design facility, flexibility, and economy, not easy—answer is, by gaining a thorough understanding of the context in which the product is used and by making sure that whatever new attributes that IT enables will enhance (and not diminish) the consumption benefits. Every designer eager to put IT to even greater use must constantly be reminded of this.

To be sure, it is not only the product designers who must be restrained and reminded. Ultimately, products are the responsibility of product managers, marketing managers, , and senior

management. It is they who must do the restraining and the reminding—first, of themselves and, then, of product designers and developers.

The message about restraining design and designers cannot be emphasized enough, for examples abound of—for want of a better phrase—IT applications run amuck:

> Every day, across America, millions of managers, bankers, doctors, teachers, chief executives, and otherwise highly competent men and women are driven to helpless frustration by the products around them. In their offices, once-familiar telephones and copiers have suddenly turned silent saboteurs, while new systems that were supposed to make work more efficient—computers, faxes, electronic mail—often do just the reverse.
>
> En route from the office, their car dashboards have a dizzying array of digital displays and their radios sport a dozen tiny buttons too small to operate. Once home, it gets worse. Stress levels soar with VCRs, CDs, message machines, electronic thermostats, keypad burglar alarms, digital clocks, microwaves, more programmable phones, and home computers. People can't seem to get things to work right anymore. Their lives have become a nightmarish world of blinks and beeps.
>
> ...
>
> Human engineering—or the lack of it—has always been a problem in some products, of course. But there's a reason why it bedevils us much more now than ever before: the microchip. ... Trouble is, too many companies wind up selling complex, overloaded gadgets that consumers can't figure out. Impenetrable manuals don't help much, either.
>
> Ricoh Co., one of Japan's leading manufacturers of office equipment, recently found in a survey of its fax customers that nearly 95% never used three key features it deliberately built into the machines to make them more appealing. ... An Ogilvy & Mather survey of VCR owners recently found that only 3% of their total TV viewing

time went to shows that were recorded in advance using the VCR's programming features. Hardly anyone used his or her VCR except to play rented movies.

The marketing implications of overwhelming consumers with all these complex and rarely used features may be more profound than companies realize. ... (*Business Week*, 1991: 58-59)

The task of "getting the product right" is, to be sure, a difficult one. For one thing, the changes that IT makes possible in products—improved information about and control over product performance, greater automation, and enhanced functionality—often affect how the user interacts with the product, and it is not always possible to predict whether and how these changes will impact the product-use experience. The difficulty is a fundamental one: in the overall context of product use, it is the human-product interface that is often the most difficult to characterize in formal, objective terms and, therefore, it is difficult to know, *ex ante*, which IT-based product-user interface construct will facilitate product use and which will detract from it.

Take a coffee machine, a thermostat control, a radio receiver, a camera, an electric cooking range, a car—just about any product you are accustomed to and comfortable with. Can you comprehensively characterize the product-user interface and lucidly detail why a particular interface construct works for you and another construct does not? You may prefer a specific information-about-and-control-over-product-performance configuration, a certain type and level of automation, and a particular mix of features and functions, but try articulating your reasons in a complete language to the product designer. It is not easy.

Now consider the use of IT to (1) alter the way you are informed about and exercise control over product performance, (2) allow you greater automation, and/or (3) offer you an expanded set of features and functions. If you had difficulty characterizing the pre-IT product-user interface and articulating which aspects of the interface worked for you and why, it will be even more difficult to predict whether the proposed IT-based changes will enhance or diminish your product-use experience.

If the individual consumer cannot reasonably predict whether the proposed IT-based changes will enhance or diminish his or her product-use experience, how can the product supplier? The risk of a possible mismatch between product-design intent and the user's product-use experience is mitigated by the fact that most products are designed not for individual consumers but for large populations of consumers. Having said that, mismatch is still possible and the cost of implementing in IT-based changes that many consumers disapprove of can be quite high.[4]

To be sure, well-crafted marketing research can provide useful insight into how a population of consumers may react to proposals for a "smarter" product,[5] but, given the highly subjective nature of many product-user interfaces, the only sure way to find out may be after consumers have had a chance to work with the IT-enhanced product and to decide for themselves whether the proposed smartness does not go far enough or if it goes too far. Of course, by then, it may be too late.

The danger of the poor predictability of market acceptance, let alone success, of IT-based changes affecting the product-user interface is that the product developer (where applicable, the product development team) will rely—perhaps even more than before—on his or her own sense of what will work. All this in a context where developer's preferences may not be transferable to the consumer population and the traditional checks and balances on the temptation to "pile on the features" have been mutated.

When incorporating IT in a product, it is not only the product-user interface that the product developer must worry about. Many products are used along with other complementary products,

---

[4]Norman (1993) reflects on the whole issue of the product-user interface in the context of increasing product smartness. Decrying the trend toward product-user interfaces that force people to adapt to the precise, repetitive, and accurate dictates of IT-laden products, he argues for a human-centered approach to the design of product-user interfaces—an approach that adapts products to the cognitive attributes that make people smart.

[5] The literature on marketing research is extensive and the applicability of the various research methodologies depends on, for example, the extent to which the smart product is new and the extent to which it is a variant of an existing not-so-smart product, whether the product is consumable or durable, the nature of the smartness (is it a new feature or function, or is it a change in the level of a feature), the stage in the product's life-cycle, and so on. For the interested reader, Churchill (1995), Dolan (1993), and Urban and Hauser (1993) may be good starting points. All three are texts and, between them, offer a useful mix of approaches and contexts.

which may or may not be produced by the same supplier and which may or may not accommodate the proposed IT-based changes in the product under discussion. In this context, the good news is that, unlike the product-user interface, the product-complementary product interface lends itself to objective characterization, and this should simplify the product development task. The bad news is that, now, there are more compatibility considerations to worry about—and, therefore, the greater the chances of getting the smart product "wrong."

The single-lens-reflex camera provides a good setting to illustrate the discussion so far. Typically, the camera itself consists of only the body; to take a picture, the "system" must also include a lens, a roll of film, and a photographer. Interactions across all the interfaces in the photographer-camera body-lens-film system are critical to good photography.

Before IT came to be incorporated in cameras and lenses, there was a shared understanding among camera suppliers and photographers about how the entire system worked. IT disrupted this understanding: there were new opportunities to collect and display more information, automate more functions, and add previously unimaginable features. Which of these enhancements would be valued by the photographer, and which would be seen as an intrusion into a satisfying photographic experience?

The question is a vexing one. Say the camera supplier can ascertain the individual photographer's preferences for information-about-and-control-over-product-performance, task automation, functionality. There remains the problem of aggregating the preferences of a population of photographers. Thus, some photographers may want aperture setting, shutter speed, focal setting, and several other pieces of information displayed in the eye piece (digitally, or in the form of an analog read-out?); while others may not care for the resulting clutter of numbers, pointers, and images. Some photographers may want automatic metering, while others may find the feature too limiting. Some photographers may like the time-and-date of photograph feature, while others may consider that a gimmick. And some photographers may value an idiot-proof camera that will do everything for them, while others may consider that demeaning. Interestingly, a whole set of

previously irrelevant consumer heterogeneities begin to play a critical role in product design: differences in the photographers' preferences for task automation, for example, had no bearing on camera-and-lens design as long as automation was not possible; they become a major design input once microelectronics and microprocessors make it possible to automate a broad (and growing) array of tasks.

We are not done yet. It is not only photographers the camera supplier must fret over. The lens, even the film, can play a critical role in determining which IT-based changes will prove to be enhancements and which will not. Thus, the camera body may be designed to load film and to sense film speed automatically, but the change will come to naught if the film does not have a special ("DX") identifier code. Similarly, the camera body may have autofocus capability, but a manual lens will only allow manual focusing; indeed, an older manual lens may not even mount on a new, autofocus-compatible camera body.

One way around the problem of getting the product right when it comes to IT-based enhancements is to offer a product line, with models varying by type and amount of information about and control over product performance, degree of automation, and extent of functionality. Indeed, the facility, flexibility, and economy that IT brings to product design can help the supplier offer an even more extensive product line than in the pre-IT world. But just because the product line can be more extensive, it does not mean that the line will be right. Precisely because of greater design facility, flexibility, and economy, the risk is that the product manager may respond to all the uncertainty by proliferating the line with a plethora of product versions and models, thus further confusing consumers, complicating the task of product management, and—quite possibly—undermining profitability.

The how-extensive-a-product-line question is an interesting one: on the one hand, the facility, flexibility, and economy that IT brings to product design may result in a proliferating product line; on the other hand, given the uncertainty of market acceptance of proposed IT-based changes, the same facility, flexibility, and economy may support market experimentation—a process that, to be

successful, warrants a dynamic, extensive product line (see Dhebar, 1995b). The emphasis on "dynamics" leads to the subject of product change—getting the product and product line right not only at any one point in time, but over an extended period of time. Here, too, the incorporation of IT in products has direct—and significant—impact on product policy.

Managing Product and Product Line Change

The challenge of getting the product and the product line right, difficult enough in the static case, takes on special significance in the dynamic setting, where, among other things, the supplier must make deliberate choices about the speed (how fast must new-and-improved versions of products follow existing versions; see Dhebar, 1995c) and the nature (how new and how improved must the new versions be; Dhebar, 1995a) of product change. In this context, not only must the product developer restrain himself or herself from "pil[ing] on the features," the product manager must also exercise restraint in the pace at which different IT-based changes are implemented in products and the resulting smart products are introduced in the market.

To see why, consider the consumer's long-term product-consumption decision. A critical issue for the consumer is the durability of his or her investment in the overall system in which the product is used. The investment durability may find the consumer far less facile and flexible than the unrestrained producer—and, therefore, less than enthusiastic about a new-and-improved version.

The most obvious source of consumer-investment durability is product durability. Durable products have a relatively long life and typically are more expensive; consumers expect to use them with satisfaction over an extended period of time. Having purchased a particular version of the product and made a not-so-easily reversible investment in it, the consumer may not be very pleased at the prospect of a (cheaper!) new-and-improved version (much new and much improved) hitting the market so soon after the existing version—this in spite of the fact that the existing version may satisfactorily serve the consumer's needs. Personal computers are an excellent example: no sooner

have you purchased a unit, took it how of the packing box, set it up, and started to use it that there is a cheaper, new-and-improved version available—and you kick yourself for not having waited.

Product durability is an important reason for consumer-investment durability, but it is not the only reason. Two other considerations, both touched on earlier, play a critical role: (1) the consumer's familiarity with the existing version of the product, including the time, the emotional energy, and the other resources expended on learning how to use the version and becoming accustomed to it; and (2) the consumer's investment in the overall system in which the existing version is used, including resources expended in acquiring and interfacing the different complementary products that are used along with the product in question. Depending on the product, these considerations can be even greater sources of consumer-investment durability than the product itself.

Earlier, I argued that it is often difficult to predict the impact of IT-based product changes on the product-user interface. The difficulty takes on special significance in the dynamic context. Most products require a period of breaking in and getting accustomed to, and the more complex the product and the more sophisticated the product-user interface (this often follows from product complexity), the greater the consumer's investment in becoming familiar with the product, let alone mastering its use. Especially for IT-rich products, the consumer's learning curve is resource-intensive, if not in terms of money, certainly in terms of time, trials-and-errors, understanding-and-memorizing command protocol, frustration, "getting used to," and so on. It is this resource intensity that contributes in a major way to consumer-investment durability: surely, the consumer can be excused for resisting a new version if a major investment must be made on becoming unaccustomed to the existing version and getting accustomed to a new version.

To be sure, many IT-based product changes—improved information about and control over product performance, greater automation, and an expanded set of features and functions—do offer added benefits. But the incremental benefits are often uncertain and distant (it may take some time before we can figure out how best to benefit from some IT–based product innovation—and we may

be disappointed), the costs of switching from the existing to the new version usually are certain and immediate, and, in any case, the expected incremental benefits may fall short of the switching costs. There's the rub: absent deliberate restraints, the facility, flexibility, and economy that IT imparts to product design can result in a quick-fire sequence of new-and-improved product versions, some or many of which may not pass the consumer's switching-benefits-greater-than-switching-costs test. The risk is a particularly acute one: IT-rich products are the very products seeing some of the most accelerated replacement of existing versions by new-and-improved versions.

The user is only one element, albeit a critical element, of the overall system in which most "smart" products are used. Many smart products require infrastructural support of other elements as well, often with electronic information transfer across the several system interfaces. Thus, autofocus cameras need autofocus lenses; electronic calculators, batteries; video-game systems, game cartridges; and personal computers, software, disks and displays. These other elements—we called them complementary products in the previous section—entail tangible (even substantial) investment on the consumer's part and contribute to the durability of his or her commitment to the existing version of the product: each time the main product in the system changes, more often than not because the use of IT is being pushed one notch further, some or all of the complementary products may need to be upgraded, and this can eat up the consumer's resources. Updating the entire system every time the main product changes can soon become a tiresome, expensive, and confusing game for the consumer, and it does not help that, increasingly, it is unclear which system element is the "main" product. Given the facility, flexibility, and economy that IT imparts to design of such products, the risk of the consumer ending up in such a game is a definite one.

None of this is to suggest that smart products should not be made smarter, or that the introduction of smarter versions should be unduly delayed. In general, there are four reasons for making products increasingly smart: supplier vision, the need to meet evolving demand, the requirement to modify the product because complementary products have become smarter, and the urge—often the imperative—to outsmart the competition. In the context of our discussion, it is in the

first and the fourth cases where the risk is the greatest of the supplier being unable to exercise necessary restraint: it is true that the supplier must exercise product leadership and vision (the Sony Walkman may not have happened otherwise), but with all the design facility, flexibility, and economy, will products get too smart too fast? As competitors avail themselves of the design facility, flexibility, and economy and fall over each other with new-and-improved (that is, smarter) versions of existing products, will the products get too smart too fast? How will less-facile, less-flexible, and vested-in-the-existing-product-version consumers deal with such rapid product change? These questions should give all those involved with product development and marketing reason to pause.

Unfortunately, there are no easy answers. By offering a product line, and not just a single version, the supplier can provide itself and its consumers greater maneuverability and additional degrees of freedom. Similarly, breaking down increased smartness into smaller, modular steps and implementing it on upgradable product architecture can be of great help. Both strategies increase the consumer's options and help mitigate the negative consequences of consumer-investment durability. On the other hand, the first strategy complicates the product-line-management effort, while the second strategy may tie the product down to antiquating architecture and possibly make a later shift even more costly. Either way, the product supplier will benefit from being intensely user-centered:

(1) always think about the overall system in which the product is used;

(2) fully understand the reasons for, and the elements of, consumer-investment durability and their time-dependent impact on the consumer's switching cost;

(3) adaptively measure the consumer benefits resulting from increased smartness; and

(4) systematically analyze the multidimensional character of demand evolution with respect to product smartness.

The Need for Many New Standards

The increasing use of IT in products raises a third product-policy concern, and that is the need for new standards with respect to the product-user and product-complementary product interfaces.

We consider the product-complementary product interface first. Perhaps the most significant impact of IT on this interface is in relation to the transfer of information: smart products like to talk to each other, and the conversation is in a whole new language. Take the example of the camera body and the lens. In older, mechanical cameras, all that mattered was that you could physically mount the lens on the camera body; the only information transfer across the interface was optical—light traveling through the lens and on to the film in the camera body. Later, as metering was automated, functional (electronic and mechanical) coupling between the body and the lens also became more important. Finally, with the introduction of microprocessors in cameras and lenses came the ability to autofocus—and, with autofocus, complex information transfer across the camera body-lens interface.

For the information transfer to be effective, the message sender and receiver should either "speak" the same language or, through a translator, understand each other's language. This requirement raises concerns about syntax, grammar, speed of communication, bandwidth, protocol, cross-talk, signal quality, and so on—not unlike the issues that any two strangers from different parts of the world have to sort out before they can effectively communicate.

The good news is that, unlike people, microprocessors can be programmed for precision and consistency: once the developers of the product and the relevant complementary products have agreed to the communications protocol and designed the interfaces appropriately, effective information transfer can be sustained over time. The bad news is that establishing the necessary protocol may take time and effort and a lot of coordination within and across industries, with each player jockeying for a position of strategic advantage. In telecommunications, computers, consumer electronics, transportation systems, and household appliances, the establishment of such standards is an important element of the larger competitive game, and the process pits business units against business units (for a long time, computer users teased IBM about how it was easier for non-IBM computers to talk with IBM personal computers and mainframes than for IBM personal computers to talk directly to IBM mainframes), companies against companies (Nintendo *vs.* Sega for home video-

game systems; video-game cartridges are not compatible across the two platforms), coalitions against coalitions (Sony-Philips and Toshiba-Matsushita-Time Warner-RCA over a format for digital videodisks[6]), and countries against countries (Japan *vs.* the United States *vs.* Europe over high-definition television).

The standards folklore is replete with such examples and, in recent years, the debate has moved from the inner sanctums of engineering associations, standards bodies, and academia to the front pages of the popular press. While the exact process varies from case to case, the issues are by now well understood:

(1) Standards for information transfer across the product-complementary product interface are essential if a smart product is to gain fast, widespread acceptance. This facilitates the development of compatible complementary products, thus offering consumers greater opportunity to "mix and match" and "plug and play."

(2) In a standards war, victory goes not necessarily to the swift, but to the player who makes the most convincing case—and can get enough consumers, complementary-product developers, and distributors to join the bandwagon so the whole process becomes self-fulfilling—that it can the first to get to a "critical mass" of adopters.

(3) Timing plays a major role in the process: a standard too soon may saddle consumers with a less-than-smart product; on the other hand, a long-drawn-out process may deprive consumers of product smartness altogether.

(4) The outcome is not dictated by technological concerns alone; indeed, in a politically charged process, the better technological solution can come a cropper.

(5) Although some standards just happen, it pays to manage the process actively and with the level of strategic management attention it deserves.

---

[6] After months of jockeying and negotiations, and under intense pressure from computer manufacturers and Hollywood movie studios, the two camps agreed on a common standard in mid-September 1995. All players were mindful of the fight over Beta and VHS formats in the late 1970s and its detrimental impact on the diffusion of a new product concept among wary consumers.

The other area where smart products often require new standards is the product-user interface. This subject does not lend itself to sophisticated economic analysis and modeling that traditional research on standards is subjected to, but, depending on the product, product-user-interface standards, it can be a much more difficult nut to crack. There are at least five reasons for this:

(1) As already argued, the product-user interface is a highly subjective construct that is not easily or comprehensively characterized.

(2) More often than not, the construct is idiosyncratic, with variations from consumer to consumer and product-use context to product-use context.

(3) People—unlike smart products talking to each other—are imprecise, inconsistent, and not easily programmable. They also cannot be designed or ordered to always conform to standards. This is not an indictment of people: these characteristics may be a disadvantage in some tasks, but they play a major role in human smartness.

(4) For product-complementary products interfaces, there are industry forums and processes to establish and enforce agreed-upon standards. Human-interface standards, in contrast, get established—if they do get established—over time as people learn to interact with different products and decide which interface they prefer over some other. The whole process is not analytical, rational, political, or deliberative; instead, it is experiential, behavioral, and adaptive, and it is based on complex trial-and-error-based individual and social learning.

(5) In any case, it in not always clear what one means by a "standard" when one talks of the product-user interface.

To illustrate the last point, take the wristwatch. The mechanical, manual-wind watch of old used gears and moving parts, the display was an analog one with two (including the second hand, three) hands circling a dial, and the watch was wound by rotating a crown. As children, we learned to read the watch's hands; growing up, we learned how to wind the watch and set the time, and we

got used to its touch-and-feel on our wrist and the tick-tick next to our ears. Now consider the digital quartz watch. It reads different, it sounds different, it needs a battery, and its many functions require the user to negotiate complicated sequences of tiny buttons. At the level of the product-user interface, how many old "standards" must be abandoned and new ones established for an effective transition from the analog mechanical watch to the digital quartz watch?

First, how the wearer tells the time is completely different. In the analog case, the wearer sees the time; in the digital case, the wearer reads it. The two activities involve very different models of information content, display, retrieval, and processing. In the case of the analog watch, a mere glance at the relative positions of the hour and the minute hands tells the wearer approximately what time it is (the approximation is sufficient for most needs); in contrast, the wearer of a digital watch must pick from, read, and intellectually process a sequence of numbers and codes displayed in one or more information windows in order to tell the time down to the fraction of a second.

Second, how the wearer controls the watch's functions is different. Thus, in the case of the analog mechanical watch, a rotation of the crown winds the watch, a pull of the crown followed by a rotation turns the hands. In contrast, the digital quartz watch needs no winding (it needs a battery, however, and replacing the battery can be a nontrivial task), and one sets the time by manipulating a specific sequence of several little buttons (the sequence may vary from brand to brand and watch model to watch model and, if the wearer has not memorized it, may need to refer to a—misplaced?—user's manual).

Then, there is the fact that the analog mechanical watch is designed and used to tell the time (and, perhaps, date and day) and to estimate time intervals. The digital quartz watch, on the other hand, can combine as a calculator, a stop-watch, a thermometer, a pulse monitor, and so on. All these functions may share the same control buttons and physical display as "time" and accessing them may require a mastery over several command sequences.

The wristwatch example illustrates the range of new-standard needs at the level of the product-user interface: changes in the form, content, density, update rate, display, and processing of

information offered by the product to the user; and changes in how the user interacts with the product to benefit from its increased automation and expanded functionality. Identifying and establishing the full set of needed standards is no mean task and can test the skills and ingenuity of the smart-product supplier. It is a crucial aspect of product policy that product managers, developers, and marketers have traditionally not needed to attend to—and, therefore, may not have the skills necessary to attend to—but must attend to.

Concluding Comments

IT helps make products smart; it also liberates product design. The liberation has its pluses and minuses. On the "pluses" side, the supplier can offer improved information about and control over product performance, greater automation, and enhanced functionality with greater design facility, flexibility, and economy; as for the minuses, what is to stop product designers and developers from piling on features and product managers from rapidly introducing and upgrading products that are smart but are difficult for the consumer to use without great confusion, cost, or adaptation? Without a deliberative product policy that recognizes the implications for product and product-line design, product and product-line dynamics, and product-use standards, the supplier may end up with a product where the consumer's cost of switching to the smart product outweighs the benefits of smartness.

I conclude my discussion as I began—with a quote on the increasing use of IT in cars. I leave it to the reader to decide on the cost-benefit tradeoffs of the resulting product smartness.

> It used to be that anyone who wanted a car with air bags, antilock brakes and a few doodads like power seats had to shop for a big Cadillac or a Mercedes. But today all those features and more are available on a Toyota Corolla—which has tested the ingenuity of luxury-car makers in dreaming up one-of-a-kind features most drivers probably don't know they need.

Hence side-mounted mirrors that automatically tilt down when the car is shifted into reverse, such as those on Infinity Q45 sedans. Admittedly, that allows a driver to see a tricycle on the driveway; but it also makes it more difficult to see a car that's passing by on the street.

No matter. This year, the Lincoln Continental is going the Q45 one better by offering automatic reverse-tilt mirrors with *memory*, meaning the car remembers which of its drivers like the feature and which don't. Trouble is, Continental drivers who aren't whizzes at operating the car's complex computer could end up staring at a glaring LED screen as big as the speedometer that reads "REVERSE MIRRORS ON OFF" (requiring the driver to choose one or the other) ...

...

[In the BMW 740i] 16 different climate-control buttons and switches ... [give] the driver and the passenger independent control over heating and air conditioning. In fact, this car is so complex—there are 11 more buttons or switches on the steering wheel—that it comes with a 34-page pocket-sized Features Operations Guide ... Says ... an automotive analyst ..., "You really should get a two-day seminar with this thing." (Lavin, 1995)

References

Business Week (1991), "I Can't Work this Thing," (April 29), 58-66.

Churchill, G. A. (1995), *Marketing Research: Methodological Foundations*, Orlando, Florida: The Dryden Press.

Dhebar, A. (1995), "Complementarity, Compatibility, and Product Change: Breaking with the Past?," *Journal of Product Innovation Management*, **12**, (March), 136-152.

__________ (1995), "Using Extensive, Dynamic Product Lines for Listening in on Evolving Demand," *European Management Journal*, **13**, (June), 187-192.

__________ (1995), "New-and-Improved High-Tech Products: Speeding Producer, Meet the Balking Consumer," Working Paper No. 3849, Sloan School of Management, Massachusetts Institute of Technology (August).

__________ (1995), "Information Technology and Product Policy (B): Information Products," Working Paper No. 3856, Sloan School of Management, Massachusetts Institute of Technology (September).

Dolan, R. J. (1993), *Managing the New Product Development Process: Cases and Notes*, Reading, Massachusetts: Addison-Wesley.

The Economist (1995), "Electronics in Cars: Get a Grip," (January 28), 76-78.

Lavin, D. (1995), "Designers Cram Luxury Cars With Gee-Whiz Gizmos," *The Wall Street Journal*, , (February 28), B1.

Norman, D. A. (1993), *Things That Make Us Smart: Defending Human Attributes In the Age of the Machine*, Reading, Massachusetts: Addison-Wesley.

Urban, G. L. and J. R. Hauser (1993), *Design and Marketing of New Products*, Englewood Cliffs, New Jersey: Prentice-Hall.

Date Due

OCT. [illegible] 1967		